

An adaptive gravitational search algorithm for global optimization

Ying-Ying Koay, Jian-Ding Tan, Chin-Wai Lim, Siaw-Paw Koh, Sieh-Kiong Tiong, Kharudin Ali
Institute of Sustainable Energy, Universiti Tenaga Malaysia (UNITEN), Malaysia

Article Info

Article history:

Received Jan 20, 2019

Revised Apr 30, 2019

Accepted May 16, 2019

Keywords:

Adaptive gravitational search

Adaptive search step

Global optimization

ABSTRACT

Optimization algorithm has become one of the most studied branches in the fields of artificial intelligent and soft computing. Many powerful optimization algorithms with global search ability can be found in the literature. Gravitational Search Algorithm (GSA) is one of the relatively new population-based optimization algorithms. In this research, an Adaptive Gravitational Search Algorithm (AGSA) is proposed. The AGSA is enhanced with an adaptive search step local search mechanism. The adaptive search step begins the search with relatively larger step size, and automatically fine-tunes the step size as iterations go. This enhancement grants the algorithm a more powerful exploitation ability, which in turn grants solutions with higher accuracies. The proposed AGSA was tested in a test suit with several well-established optimization test functions. The results showed that the proposed AGSA out-performed other algorithms such as conventional GSA and Genetic Algorithm in the benchmarking of speed and accuracy. It can thus be concluded that the proposed AGSA performs well in solving local and global optimization problems. Applications of the AGSA to solve practical engineering optimization problems can be considered in the future.

Copyright © 2019 Institute of Advanced Engineering and Science.
All rights reserved.

Corresponding Author:

Ying-Ying Koay,

Institute of Sustainable Energy,

Universiti Tenaga Malaysia (UNITEN), Malaysia.

Email: tjinding@uniten.edu.my

1. INTRODUCTION

Optimization problems in the practical world are getting more and more complex and difficult. Artificial intelligence and soft computing algorithms has become one of the most popular means to solve increasingly high-dimensional global optimization problems [1]. Literature study shows rapid developments on evolutionary and population-based optimization algorithms over the past few decades. These algorithms have proven to be very useful in solving various types of mathematical and practical optimization problems [2-6]. Inspired by biological features, natural events, and scientific-related effects, many of these algorithms shows very promising performance in solving complex global optimization problems. Among others is the Gravitational Search Algorithm.

Gravitational Search Algorithm (GSA) is a relatively new population-based global optimization algorithm [1]. In GSA, all the solutions are considered as agents, moving in the space in the feasible solution range. Inspired by the Newton's law of gravity [7], the objective value of an agent correlate with its mass. Since a heavier mass yields a higher attraction force in the law of gravity, a better objective function in the GSA gives a relatively higher mass value to a particular agent. This, then, in turn, causes the agents in the search space to move towards agents with better objective yields [8-9]. This mechanism allows the agents to learn from the search results and information of each other before the moving in a particular iteration. It is essentially a learning strategy in nature which grants GSA the ability to explore for a more diversified solutions. However, this fully-informed learning strategy can cause a poor trade-off between the exploration

and the exploitation abilities of the algorithm [7]. Over the past few decades, many different enhancements are proposed and tested for different population-based heuristic search mechanisms. Some of the examples can be found in [10-15]. Similarly, modifications and improvements on the GSA can also be found in the literature. Like many other population-based algorithm, one of the most commonly found modification is on the introduction of new operators into a conventional GSA. For example, a disruption operator was proposed to further explore the search space [16]. Other researchers attempted an opposition-based learning in the initialization stage of the search [17]. The Black Hole Theory was also introduced into the GSA in order to prevent premature convergence [18]. Beside all these, many other application-specific modifications on the GSA can also be found in the literature [19-24].

In this paper, an adaptive search step local search mechanism is introduced into the conventional GSA. The proposed modification is designed to enhance the exploitation ability of the GSA. It enables the algorithm to analyse the information around the neighbourhood of an agent, which in turn leads to solutions with relatively higher accuracies. This paper consists of four chapters. Chapter 2 discusses the mechanism of the conventional GSA and the proposed modification in details. The results from the conducted experiments are presented and analysed in Chapter 3. Chapter 4 offers a comprehensive conclusion drawn from the outcomes of this research.

2. ADAPTIVE GRAVITATIONAL SEARCH ALGORITHM

A conventional GSA essentially consists of several stages, including mass calculation, acceleration calculation, velocity calculation and agent displacements. Similar to most of the population-based optimization algorithms, the GSA begins with the initialization stage. In the initialization stage, a pre-fixed number of solutions are randomly selected from the feasible solution search-space. These solutions are known as agents in the GSA. Each agent is randomly assigned a feasible value from each dimension. Upon determining the yielded fitness function of each agent, the algorithm moves on to the looping stage, in which the algorithm will loop until a pre-determined termination criteria is met.

The looping stage of a conventional GSA begins with the total gravitational force calculation. For each of the agents, $X = [x_{t1}, x_{t2}, \dots, x_{tj}, \dots, x_{tD}]$, formula (1) is used to calculate the gravitational force from the K best agents [1].

$$F_{ij}^t = \sum_{k \in KBest, k \neq i} rand_k \cdot G^t \cdot \frac{M_k^t \cdot M_i^t}{R_{i,k} + \epsilon} \cdot (x_{kj}^t - x_{ij}^t) \tag{1}$$

in which $i = 1, 2, \dots, NP$ and $j = 1, 2, \dots, D$. t indicates the current generation; D denotes the dimension of the problem; $rand_k$ is a random step value between 0 to 1. M_i and M_k denote the masses of agents i and k respectively; $R_{i,k}$ represents the distance between agents i and k . $KBest$ is the group of the top K best agents with best fitness yields, in which is defined in formula (2) [13].

$$K = NP - (NP - 1) \times \frac{t}{t_{max}} \tag{2}$$

G^t in formula (1) denotes the gravitational coefficient. This coefficient can be calculated using formula (3) [25].

$$G^t = G^0 \times \exp\left(-\tau \times \frac{t}{t_{max}}\right) \tag{3}$$

It is suggested in [13] to set $G^0 = 100$ and $\tau = 20$. t_{max} is the maximum generation number. The mass calculation for an agent can be calculated using (4) and (5) [25].

$$M_i^t = \frac{q_i^t}{\sum_{k=1}^{NP} q_k^t} \tag{4}$$

$$q_i^t = \frac{f(X_i^t) - f(X_{Worst}^t)}{f(X_{Best}^t) - f(X_{Worst}^t)} \tag{5}$$

X_{Worst}^t represents the worst agent with the worst objective yield. X_{Best}^t , on the other hand, is the best agent with the best fitness yield. Upon completing the total gravitational force calculations, the algorithm moves on to calculate the acceleration of each agent. This can be carried out using (6) [25].

$$a_{i,j}^t = \frac{F_{i,j}^t}{M_i^t} = \sum_{k \in K \text{ Best}, k \neq i} rand_k \cdot G^t \cdot \frac{M_k^t}{R_{i,k}^t + \epsilon} \cdot (x_{k,j}^t - x_{i,j}^t) \tag{6}$$

With the acceleration calculated, the velocity of each agent can now be determined, using (7) [25]. Finally, the agents are displaced and the algorithm loops on.

$$v_{i,j}^{t+1} = rand_i \times v_{i,j}^t + a_{i,j}^t \tag{7}$$

$$x_{i,j}^{t+1} = x_{i,j}^t + v_{i,j}^{t+1} \tag{8}$$

In this research, a modification is proposed with the aim to enhance the GSA with a stronger exploitation ability. A local search mechanism with adaptive search step size is designed for the purpose of this modification. This search mechanism is designed to search the neighbourhood of a particular agent with a relatively larger search step. As the iterations go, the search step size is then automatically adjusted for a more detailed search. This adaptive search step feature enables the algorithm to perform a more scrutinized search without wasting too much time at the beginning of the search, which in turn grants solutions with higher accuracies, quicker. Figure 1 shows a better illustration of the decision making flow of the search mechanism, where λ denotes the search step, D_i refers to the solution value of an agent in a particular dimension at iteration i , while λ_N and λ_C indicate the new search step for the next iteration and the current search step, respectively.

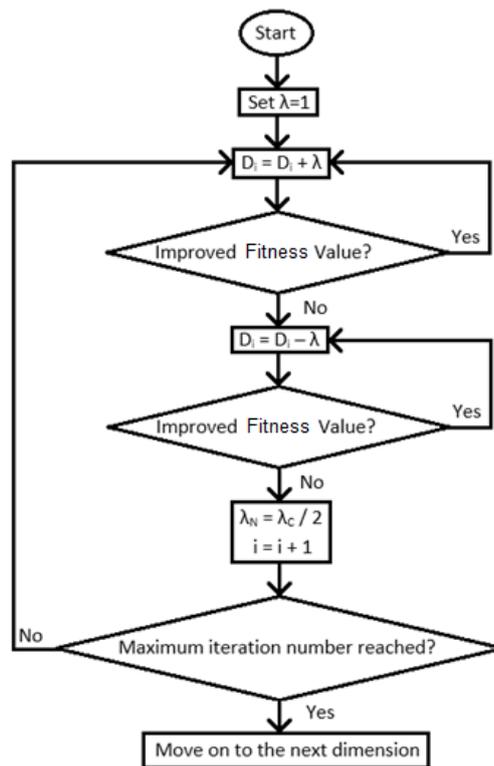


Figure 1. The proposed adaptive search step local search mechanism

The proposed adaptive search step local search mechanism is then inserted into the looping stage of the GSA before the calculation of new total gravitational force begins, making it a new Adaptive Gravitational Search Algorithm (AGSA). Table 1 shows the inserted adaptive search step local search mechanism in the AGSA.

Table 1. Algorithmic Steps of the Proposed AGSA

```

Adaptive Gravitational Search Algorithm
t = 0;
FEs = 0;
Create a random initial population;
while FEs < MAX_FEs do
for i = 1 to NP do
Adaptive search step local search mechanism
Obtain the total force that exerts on the ith solution according to (1);
Calculate the acceleration of the ith solution according to (6);
Compute the new velocity of the ith solution according to (7);
Obtain the new position of the ith solution according to (8);
Evaluate of the ith solution;
if Xt i is better than GBX then
GBX = Xt i ;
end if
FEs = FEs + 1;
end for
t = t + 1 ;
end while
    
```

3. RESULTS AND ANALYSIS

Experiments are carried out in order to investigate the performance of the proposed algorithm in solving optimization problems. The proposed AGSA is put to test to solve several well-established optimization test functions found in the literature. Table 2 shows the details of the test suites employed. The Rastrigin and Ackley test functions provide more rigorous tests on the global search ability of the algorithms as these functions come with a single global optima point with multiple local traps. All the test functions are minimization problems.

Table 2. The Test Suite

Function	Formulation	Range
Sphere	$f_1(x) = \sum_{i=1}^n x_i^2$	[-500,500]
Rastrigin	$f_8(x) = \sum_{i=1}^n (x_i^2 - 10\cos(2\pi x_i) + 10)$	[-5,5]
Ackley	$f_9(x) = -20 \cdot \exp\left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}\right) + 20 - \exp\left(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)\right) + e$	[-32, 32]
Rosenbrock	$f_2(x) = \sum_{i=1}^n (100(x_i^2 - x_{i+1})^2 + (x_i - 1)^2)$	[-2,2]
Booth	$f_1(x) = (x_1 + 2x_2 - 7)^2 + (2x_1 + x_2 - 5)^2$	[-32, 32]

In order to benchmark the difference and improvement of the proposed AGSA, the results are compared to that of a conventional GSA. Genetic Algorithm (GA), a well-established optimization algorithm is also included in the benchmarking to further clarify the stand of the AGSA in terms of performance compared to other optimization algorithms from the literature.

The tests are conducted using a 64-bit Intel Core-5 processor with 4 gigabytes or RAM. The algorithms are set to begin with 10 initial solutions. To avoid stochastic discrepancies, 10 independent runs are carried out for each of the algorithms in solving each of the test functions. The average values of the solutions returned are shown in Table 3. The best results in the comparison are highlighted in bold face.

It can be observed from Table 3 that the AGSA has outperformed the conventional GSA in all the test functions in term of solutions accuracy as AGSA successfully returned with solutions with relatively lower average objective values. GA gave no competitive results, especially in test functions with multiple optima points, such as the Rastrigin and Ackley tests. Some convergence process are sampled from the experiments to examine the convergence performance of each of the algorithms. Figure 2 shows an example of convergence process comparison between the three algorithms in solving the Sphere test function. It can be observed that the modified AGSA shows a faster and more rapid convergence process compared to the other algorithms. The AGSA also managed to reach a relatively lower fitness value at the end of the iterations.

Table 3. Comparison on the Average Values of the Returned Solutions

	AGSA	GSA	GA
Sphere	6.34E-06	3.23E-03	1.79E-02
Rastrigin	9.11E-07	1.27E-04	1.04E+00
Ackley	7.65E-06	5.52E-03	1.88E+00
Rosenbrock	5.01E-07	7.40E-04	3.94E-02
Booth	5.90E-07	5.13E-05	9.81E-02

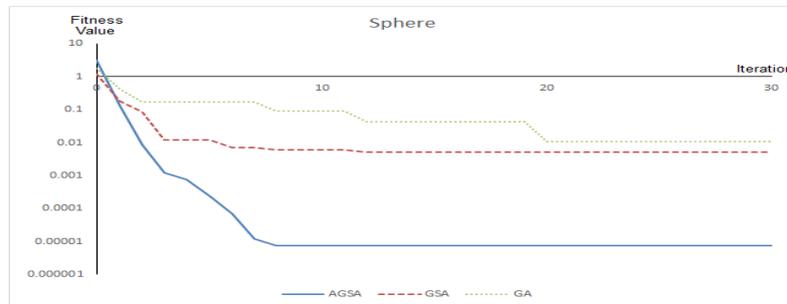


Figure 2. Convergence process samples comparison in solving Sphere test problem

The convergence performance and efficiency can vary in solving test functions with single and multiple optima points. To examine the performance of the algorithms in solving test problems with multiple optima points and local optima traps, some convergence process in solving the Rastrigin problem are sampled and compared in Figure 3.

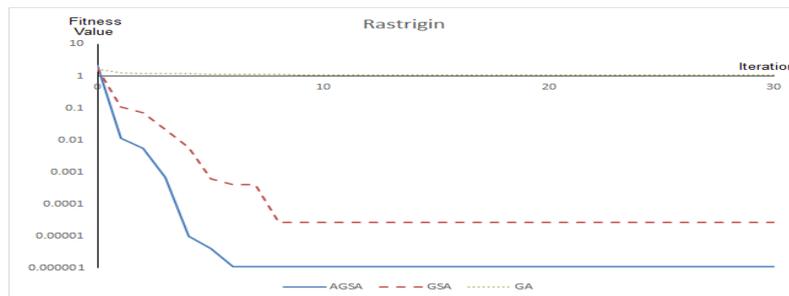


Figure 3. Convergence process samples comparison in solving test problem with multiple local optima traps

Note that the Objective Value axis for both Figure 2 and 3 are in logarithmic scales. It can be observed from both figures that the AGSA shows rapid objective value improvements at the beginning stages of the convergence processes. The larger search steps enabled the algorithm to search in relatively quicker speed. It can also be observed from the graphs that the AGSA found relatively lower objective values, which indicates that it outperformed other algorithms in finding solutions with relatively higher accuracies. The adaptive search steps enabled AGSA to further scrutinized the search for better solutions in the proximity of a pre-found solution. This gives a boost in solving the speed and accuracy problems in a conventional GSA.

4. CONCLUSION

In this research, an Adaptive Gravitational Search Algorithm (AGSA) is proposed. The AGSA incorporate an adaptive search step local search mechanism into a conventional GSA. This modification enhanced the algorithm with a more powerful exploitation ability, which in turn grants solutions with higher accuracies. The proposed AGSA is tested in a test suit with several well-established optimization test functions, including the Rastrigin function, Ackley function, Sphere function, Booth function and Rosenbrock function. The results show that the proposed AGSA out-performed other algorithms in the

benchmarking and comparison. It can thus be concluded that the proposed AGSA performs well in solving local and global optimization problems. With some minor adjustment, the implementation of AGSA can be considered to solve actual engineering optimization problems in time to come.

ACKNOWLEDGEMENTS

The authors express great acknowledgement to Universiti Tenaga Nasional (UNITEN), Malaysia for the support of this research under internal grant J510050689.

REFERENCES

- [1] G. Sun, A. Zhang, Z. Wang, Y. Yao, J. Ma, Gary Douglas Couples. "Locally informed gravitational search algorithm". *Knowledge-Based Systems*, vol. 104, pp. 134-144, 2016.
- [2] S.P. Brooks, B.J. Morgan. "Optimization using simulated annealing", *Stat*, pp. 241-257, 1995.
- [3] R. Chelouah, P. Siarry. "Genetic and nelder mead algorithms hybridized for a more accurate global optimization of continuous multimimima functions". *Eur. J. Oper. Res.* vol. 148(2), pp. 335-348, 2003.
- [4] P. Civicioglu, "Transforming geocentric cartesian coordinates to geodetic coordinates by using differential search algorithm". *Comput Geosci*, vol. 46, pp. 229-247, 2012.
- [5] P. Civicioglu. "Backtracking search optimization algorithm for numerical optimization problems", *Appl. Math. Comput*, vol. 219(15), pp. 8121-8144, 2013.
- [6] M. Dorigo, V. Maniezzo, A. Colorni. "Ant system: optimization by a colony of co-operating agents", *IEEE Trans. Syst., Man, Cybern., Part B: Cybern.*, vol. 26 (1), pp. 29-41, 1996.
- [7] E. Rashedi, H. Nezamabadi-Pour, S. Saryazdi, "GSA: a gravitational search algorithm", *Inf. Sci.* vol. 179(13), pp. 2232-2248, 2009.
- [8] S. Mirjalili, S.Z.M. Hashim. "A new hybrid PSOGSA algorithm for function optimization", in: International Conference on Computer and Information Application (ICCIA), IEEE, pp. 374-377, 2010.
- [9] W. Zhang, P. Niu, G. Li, P. Li. "Forecasting of turbine heat rate with online least squares support vector machine based on gravitational search algorithm". *Knowl.-Based Syst*, vol. 39, pp. 34-44, 2013.
- [10] M. Rasheed, R. Omar, M. Sulaiman, W.A. Halim. "Particle swarm optimisation (PSO) algorithm with reduced number of switches in multilevel inverter (MLI)". *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 14(3), pp. 1114-1124, 2019.
- [11] J.D. Tan, M. Dahari, S.P. Koh, Y.Y. Koay, I.A. Abed. "A new experiential learning electromagnetism-like mechanism for numerical optimization". *Expert Systems with Applications*, vol. 86, pp. 321-333, 2017.
- [12] J.D. Tan, M. Dahari, S.P. Koh, Y.Y. Koay, I.A. Abed. "An improved electromagnetism-like algorithm for numerical optimization". *Theoretical Computer Science*, vol. 641, pp. 75-84, 2016.
- [13] J.D. Tan, M. Dahari, S.P. Koh, Y.Y. Koay, I.A. Abed. "Analysis of the Effect of Search Step Size on the Accuracy and Convergence Properties of Electromagnetism-Like Mechanism Algorithm". *Journal of Multi-Valued Logic & Soft Computing*, vol. 28, pp. 429-441, 2016.
- [14] M.A. Abd Rahman, B. Ismail, K. Naidu, M.K. Rahmat. "Review on population-based metaheuristic search techniques for optimal power flow". *Indonesian Journal of Electrical Engineering and Computer Science (IJECS)*, vol. 15(1), pp. 373-381, 2019.
- [15] D.B. Mishra, A.A. Acharya, R. Mishra. Evolutionary algorithms for path coverage test data generation and optimization: a review. *Indonesian Journal of Electrical Engineering and Computer Science (IJECS)*, vol 15(1), pp. 504-510, 2019.
- [16] S. Sarafrazi, H. Nezamabadi-Pour, S. Saryazdi, "Disruption: a new operator in gravitational search algorithm", *Scientia Iranica*, vol. 18(3), pp. 539-548, 2011.
- [17] B. Shaw, V. Mukherjee, S. Ghoshal, "A novel opposition-based gravitational search algorithm for combined economic and emission dispatch problems of power systems". *Int. J. Electr Power Energy Syst*, vol. 35(1), pp. 21-33, 2012.
- [18] M. Doraghinejad, H. Nezamabadi-pour. "Black hole: A new operator for gravitational search algorithm". *Int. J. Comput. Intell. Syst*, vol. 7(5), pp. 809-826, 2014.
- [19] F. Ghorbani, H. Neamabadi-Pour. "On the convergence analysis of gravitational search algorithm". *J. Adv. Comput. Res*, vol. 3(2), pp. 45-51, 2012.
- [20] C. Lopez-Molina, H. Bustince, J. Fernandez P. Couto, B. De Baets. "A gravitational approach to edge detection based on triangular norms", *Pattern Recognit*. vol 43, pp. 3730-3741, 2010.
- [21] W.E. Wright, "Gravitational clustering", *Pattern Recognit*, vol 9, pp. 151-166, 1977.
- [22] S. Mirjalili, S.Z.M. Hashim, H.M. Sardroudi. "Training feed forward neural networks using hybrid particle swarm optimization and gravitational search algorithm". *Appl. Math. Comput*, vol. 218, no. 22, pp. 11125-11137, 2012.
- [23] S. Yazdani, H. Nezamabadi-pour, S. Kamyab. "A gravitational search algorithm for multimodal optimization". *Swarm Evol. Comput*, vol 14, pp. 1-14, 2014.
- [24] X. Li, M. Yin, Z. Ma. "Hybrid differential evolution and gravitation search algorithm for unconstrained optimization". *Int. J. Phys. Sci* 6, vol. 25, pp. 5961-5981, 2011.
- [25] E. Rashedi, E. Rashedi, H. Nezamabadi-pour. "A comprehensive survey on gravitational search algorithm". *Swarm and Evolutionary Computation*, vol. 4, pp. 141-158, 2018.